Edwards w-coordinate를 이용한 SIDH 구현[1)]

김수리*, 윤기순**, 박영호***, 홍석희*
*고려대학교 정보보호대학원
**NSHC Inc.
***세종사이버대학교 정보보호과
e-mail: suhrikim@gmail.com

Implementation of SIDH using Edwards' w-coordinate

Suhri Kim*, Kisoon Yoon*, Young-Ho Park***, Seokhie Hong*
*Center for Information Security Technologies (CIST), Korea University
**NSHC Inc.
***Sejong Cyber University

요 약

본 논문에서는 Edwards 곡선의 w-coordinate를 이용한 isogeny로 SIDH를 구현하는 방법에 대해 처음으로 제안한다. 이를 위해, 먼저 division polynomial을 이용하여 기존 제안된 w-coordinate 3-isogeny를 4M+6S에서 2M+3S로 연산량을 감소시켜 50%의 속도 향상을 달성하였다. 또한, w-coordinate isogney공식을 사용하기 위해 새로 유한체를 제안하였으며 안전성을 분석하였다. 본 논문의 결과 Edwards 곡선을 이용한 SIDH는 Montgomery 곡선과 비해 2.29% 이상의 속도 향상을 예상할 수 있다.

1. 서론

현재 전자서명이나 공인인증에 사용되는 공개키 암호인 RSA나 ECC의 안전성은 수학적 난제에 기반을 둔다. RSA는 인수분해의 어려움에 기반을 두고 있으며 ECC는 타원곡선 위에 정의된 이산대수 문제의 어려움에 기반을 두고 있다. 하지만 최근 양자 컴퓨터의 개발이 가시화됨에 따라 Shor 알고리즘이 구현될 가능성이 커지면서 이에 대응하는 새로운 공개키 암호인 후 양자 암호(Post-quantum cryptography, PQC)에 대한 연구가 활발히 진행되고 있다. PQC는 고전 컴퓨터에 구현할 수 있지만, 아직 효율적인 양자 알고리즘이 개발되지 않아 양자 컴퓨터에도 안전한 알고리즘을 의미한다. PQC는 다변수 기반, 래티스 기반, 코드 기반, isogeny기반, 해시 기반 전자서명의 5가지로 분류되고 있다. 현재 NIST에서는 양자 컴퓨팅 시대에 사용할 안전한 공개키 암호를 위한 PQC 표준화 공모를 2017년부터 시작하고 있으며, 현재는 round 2단계이다.

Isogeny 기반 암호는 두 타원곡선 사이의 isogeny를 찾는 어려움에 기반을 둔 암호 시스템이다. 2011년 De Feo와 Jao가 제안한 SIDH (Supersingluar Isogeny Diffie-Hellman)는 supersingular 곡선을 이용한 Diffie-Hellman 방식의 키 교환 시스템으로, 다른 PQC 암호에 비해 작은 키 사이즈를 가지는 장점이 있다 [7]. SIDH를 확장한 SIKE는 현재 NIST 표준화 공모전 round 2 후보이다.

Isogeny 기반 암호는 다른 PQC 암호에 비해 작은 키 사이즈를 가지는 장점이 있지만, 다른 암호에 비해 느리다는 단점을 가지고 있다. 따라서 효율적인 구현을 중심으로 연구가 진행되고 있으며, 현재 state-of-the-art 구현은 임의의 타원곡선에서 타원곡선 연산 및 isogney 연산이 효율적인 Montgomery 곡선을 이용해 구현되어 있다 [5]. 하지만 다른 타원곡선을 사용할 경우의 효율성에 관한 연구가 진행되고 있다. Meyer 등은 [11]에서 isogeny 연산은 Montgomery 곡선에서, 타원곡선 연산은 twisted Edwards 곡선을 이용하는 hybrid 방법을 제안하였다. 하지만 Bos 등은 해당 방법을 비효율적이며, twisted Edwards 곡선 사용은 isogeny 기반 암호에 비효율적이라는 결론을 내렸다 [2]. 그러나 Kim 등은 SIDH에 사용하는 유한체 특성상 twisted Edwards 곡선이 아닌 Edwards 곡선을 이용할 수 있고, Edwards 곡선에서 isogeny 공식을 제안하였다 [8]. 이후 [9]에서 Edwards isogeny 공식과 Montgomery 공식을 합성하여 새로운 hybrid 방법을 제안하였고, 이는 기존 SIDH 보다 빠른 속도를 가진다. 한편 [10]에서는 [9]에서 제안한 w-coordinate를 이용해 Edwards 위에서 isogeny 공식을 유도하면 타원곡선 계수 복원과정이 더 효율적이라는 결과를 도출했다. 그들은 해당 방법을 홀수 차수 isogney만 사용하는 키 교환 알고리즘인 CSIDH (Commutative SIDH)에 적용하였고 이전 결과보다 속도가 20% 향상되었다 [3, 10].

본 논문에서는 [10]에서 제안한 Edwards 공식을 SIDH

1) "본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음" (IITP-2019-2015-0-00385)

환경에 적용해 보고 기존 Montgomery 속도와 비교하였다. 이를 위해 먼저 [10]에 제안한 3-isogeny 공식을 division polynomial을 이용해 최적화하였으며, 기존 4M+6S의 연산량에서 2M+3S의 연산량으로 50% 속도 향상을 달성하였다. 또한, 홀수 차수를 이용하기 위해 새로운 유한체를 정의하였으며 안전성을 분석하였다. 본 논문의 결과 Edwards 곡선을 사용할 경우 Montgomery 곡선보다 2.29% 이상의 속도 향상을 예상할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 타원곡선의 형태와 SIDH에 대해 소개를 한 뒤, 3장에 w-coordinate를 이용한 타원곡선 연산식 및 3-isogeny 최적화 공식을 제안한다. 그 후 새로운 유한체를 위한 소수를 제안하고 안전성을 분석한 뒤, SIDH를 구현 시 연산량을 비교한다.

2. 배경지식

1) Forms of Elliptic Curves

K를 characteristic이 2나 3이 아닌 field라 하자. K위에서 정의된 타원곡선은 distinguished point을 가지는 genus 1의 smooth projective algebraic curve이다. 타원곡선 위의 점들과 무한원점이라 표현되는 distinguished point는 덧셈 그룹을 형성한다. 타원곡선을 나타내는 식에 따라 타원곡선을 분류하고 있으며, 본 논문에서는 Montgomery 곡선과 Edwards 곡선에 관해 설명한다.

① Montgomery 곡선

$$M_{A,B}: By^2 = x^3 + Ax^2 + x,\ b(a^2-4) \neq 0$$

으로 나타내지는 곡선을 Montgomery 곡선이라 한다. Montgomery 곡선의 j-invariant는

$$j(M_{A,B}) = 256(A^2-3)^3/(A^2-4)$$

로 정의한다.

② Edwards 곡선

$$E_d: x^2+y^2 = 1+dx^2y^2,\ d \neq 0,1$$

으로 나타내지는 곡선을 Edwards 곡선이라 한다. Edwards 곡선의 j-invariant는

$$j(E_d) = 16(1+14d+d^2)^3/d(1-d)^4$$

로 정의된다. E_d가 항상 위수가 4인 점을 가진다는 점이 타원곡선 암호에서의 사용을 제한하기 때문에 Bernstein 등은 twisted Edwards 곡선을 제안하였으며, twisted Edwards 곡선은 다음과 같이 나타내진다.

$$E_{a,d}: ax^2+y^2 = 1+dx^2y^2,\ a,d \neq 0$$

③ Montgomery 곡선과 (twisted) Edwards 곡선의 관계

Bernstein 등은 [1]에서 K위에서 정의된 모든 twisted Edwards 곡선은 Montgomery 곡선과 birationally equivalent 하다는 것을 증명하였다. a,d를 K에서 0이 아닌 원소라 하자. 이 때, $E_{a,d}$는 다음과 같은 map에 의해 $M_{A,B}$와 birationally equivalent 하다.

$$(x,y) \to (u,v) = \left(\frac{1+y}{1-y}, \frac{1+y}{(1-y)x}\right)$$

여기에서 $A = 2(a+d)/(a-d)$이고 $B = 4/(a-d)$이다.

2) Isogeny 와 Velu의 공식

두 타원곡선 E_1과 E_2사이의 isogeny는 커널의 원소가 유한개인 surjective group homomorphism이다. 유한체 K 위에 두 타원곡선이 isogenus 하다는 의미는 $\phi: E_1 \to E_2$를 만족하는 isogeny ϕ가 존재한다는 것을 의미한다. 또한, 만약 타원곡선의 유한 subgroup G가 주어진다고 할 때, $E' = E/G$를 만족하는 타원곡선 E'가 존재하며, 커널이 G인 isogeny $\phi: E \to E'$가 존재한다.

두 타원곡선 사이의 isogeny를 구성하는 방법은 2가지가 존재하며, 하나는 Velu가 주어진 타원곡선과 subgroup으로 isogeny를 구하는 방법이고, 후에 Kohel이 kernal polynomial로 isogeny를 구하는 방법을 제안하였다.

3) SIDH

SIDH를 이용한 키 교환과정은 다음과 같다. 먼저 서로소인 두 정수 ℓ_A와 ℓ_B를 설정하고, cofactor f를 설정하여 소수 $p = \ell_A^{e_A}\ell_B^{e_B}f \pm 1$를 선택한다. F_{p^2}에서 정의된 supersingular 타원곡선 E를 선택하면 이 타원곡선의 위수는 $(\ell_A^{e_A}\ell_B^{e_B}f)^2$를 만족한다. 이 경우 $\ell \in \{\ell_A, \ell_B\}$와 $e \in \{e_A, e_B\}$에 대해 ℓ^e-torsion subgroup을 생각할 수 있다. Alice는 $\ell_A^{e_A}$-torsion 그룹을 생성하는 기저$\{P_A, Q_A\}$를 선택하고 Bob은 $\ell_B^{e_B}$-torsion 그룹을 생성하는 기저 $\{P_B, Q_B\}$를 생성한다. Alice는 개인키 $m_A, n_A \in Z/\ell_A^{e_A}Z$를 랜덤하게 선택하고 커널 $\langle R_A \rangle = \langle [m_A]P_A + [n_A]Q_A \rangle$을 생성한다. 그 뒤 Velu 공식을 이용하여 E에서 $E_A = E/\langle R_A \rangle$로 가는 isogney ϕ_A를 계산한다. Alice는 Bob에게 $(E_A, \phi_A(P_B), \phi_A(Q_B))$를 전달하고 Bob도 Alice와 같은 방법으로 연산한 뒤 $(E_B, \phi_B(P_A), \phi_B(Q_A))$를 Alice에게 전달한다. Alice는 그 뒤 $\langle R'_A \rangle = \langle [m_A]\phi_B(P_A) + [n_A]\phi_B(Q_A) \rangle$를 연산하고 Velu 공식을 이용해 $E_{AB} = E_B/\langle R'_A \rangle$를 연산한다. Bob도 같은 방법으로 $E_{BA} = E_A/\langle R'_B \rangle$를 연산한다. Alice와 Bob 사이의 공유된 비밀키는 $j(E_{AB}) = j(E_{BA})$로 한다.

3. Edwards 곡선을 이용산 SIDH

이 장에서는 Edwards 곡선에서 w-coordinate과 [10]에서 제안한 isogeny 공식을 이용하여 SIDH를 구현할 때 연산량을 비교한다. 먼저 w-coordinate를 소개한 뒤, 최적화된 3-isogeny 연산 방법을 제시한다. 마지막으로 새로

운 유한체를 선택해 안전성을 분석한 뒤, Montgomery 곡선과 Edwards 곡선에서 연산량을 비교한다.

1) w-coordinate를 이용한 연산 비교

① w-coordinate를 이용한 타원곡선 연산

Farashahi 와 Hosseini는 [6]에 w-coordinate를 이용해 Edwards 곡선에서 타원곡선의 연산을 정의하는 방법을 제안했다. 타원곡선 위의 점 (x,y)에 대해서 rational map을 $w(x,y)=dx^2y^2$ 또는 $w(x,y)=x^2/y^2$으로 정의한다. 두 경우 같은 타원곡선 연산 공식이 도출되기 때문에 $w(x,y)=dx^2y^2$ 경우에만 간략히 설명한다.

E_d위의 점 $P=(w,y)$라 할 때, $-P=(-x,y)$가 되어 $w(P)=w(-P)$가 되고, $w(O)=0$을 만족한다. $P_1=(x_1,y_1)$, $P_2=(x_2,y_2)$가 E_d위의 두 점이라 하자. $w_0=(2P_1)$, $w_3=w(P_1+P_2)$, $w_4=w(P_1-P_2)$라 하면 E_d 위에서 doubling과 differential addition은 다음과 같이 정의된다.

$$w_0=\frac{4w_1((w_1+1)^2-ew_1)}{(w_1^2-1)^2},\ e=4/d \qquad (1)$$

$$w_3w_4=\frac{(w_1-w_2)^2}{(w_1w_2-1)^2}$$

SIDH 환경에서는 구현의 효율성을 위해 타원곡선 위의 점뿐만 아니라 타원곡선 계수도 projective coordinate를 사용한다. $P=(x,y)$를 E_d위의 점이라 하자. $d=D/C$, $w=dx^2y^2$, $w=W/Z$라 하면 $P=(W:Z)$에 대해 [2]$P=(W':Z')$는 식 (1)을 이용하면

$$W'=4WZ(D(W+Z)^2-4CWZ)$$

$$Z'=D(W+Z)^2(W-Z)^2$$

와 같으며, 4M+2S의 연산량이 든다. 여기서 M은 유한체 위의 곱셈, S는 유한체 위의 제곱 연산을 의미한다.

② w-coordinate를 이용한 isogney 공식

Kim 등은 [10]에서 w-coordinate를 이용한 홀수 차수 isogeny 공식을 제안하였다. $\ell=2s+1$라 하자. 타원곡선 E_d에 대해 커널을 $\langle P\rangle=\{(0,1),(\pm\alpha_1,\beta_1),\ldots,(\pm\alpha_s,\beta_s)\}$로 정의한다. $w_i=d\alpha_i^2\beta_i^2$라 할 때, E_d에서 $E_{d'}=E_d/\langle P\rangle$로 가는 ℓ-isogeny는 다음과 같이 정의된다.

$$\phi(w)=w\prod_{i=1}^{s}\frac{(w-w_i)^2}{(1-ww_i)^2},\ d'=d^{\ell}\prod_{i=1}^{s}\frac{(w_i+1)^8}{4^4} \qquad (2)$$

2) 효율적인 3-isogeny

$P=(\alpha_1,\beta_1)\in E_d$를 3-torsion point라 하고, $w(P)=w_1$, $w_1=W_1/Z_1$라 하자. P를 커널로 하는 E_d에서 $E_{d'}=E_d/\langle P\rangle$로 가는 3-isogney를 ϕ라 하자. E_d위의 점 $(W:Z)$에 대해 $(W':Z')=\phi(W:Z)$는 다음과 같이 연산된다.

$$W'=W(WZ_1-ZW_1)^2,\quad Z'=Z(WW_1-ZZ_1)^2$$

$$D'=D^3(W_1+Z_1)^8,\quad C'=C^3(W_1+Z_1)^8$$

위 식에서 $d=D/C$, $d'=D'/C'$으로 W', Z'의 연산량은 4M+2S, D', C'의 연산량은 4M+6S가 소요된다. 하지만 division polynomial을 이용하면 D', C'의 연산을 최적화할 수 있다. n차 division polynomial은 n-torsion point를 근으로 가진다. 이를 이용하면 타원곡선의 계수를 torsion point로 표현할 수 있다. 식 (2)에 3차 division polynomial을 이용해 $d=(16w_1)/(6w_1^2+8w_1+3-w_1^4)$를 대입하면 D', C'는 다음과 같이 연산된다.

$$D'=-16W_3^2Z_3$$

$$C'=(W_3^2-2W_3Z_3-3Z_3^2)(W_3^2-6W_3Z_3+9Z_3^2)$$

이 경우, D', C'의 연산량은 2M+3S가 된다.

3) 제안하는 파라미터

기존 SIDH에서는 ℓ_A, ℓ_B를 각각 2, 3을 사용하고 있다. 하지만 [10]에서 제안한 공식과 본 논문에서 최적화한 3-isogeny의 성능을 확인하기 위해 홀수만을 이용한 소수의 제안이 필요하다. 본 논문에서는, 양자 컴퓨팅 환경에서 128비트 보안강도를 가지는 파라미터를 다음과 같이 제안한다.

<표 1> SIKE 파라미터와 논문 파라미터 비교

종류	소수	비트
SIKE751	$p=2^{372}3^{239}-1$	751
OURS	$p=3^{227}5^{168}\cdot 28-1$	755

위 파라미터의 안전성을 분석하기 위해 먼저 isogeny 암호의 기반문제를 알아본다. Isogeny 기반 암호의 안전성은 DSSI와 CSSI의 어려움에 기반을 두며, 정의는 다음과 같다.

- DSSI (Decisional Supersingular Isogeny Problem) F_{p^2} 위에 타원곡선 E_0가 주어졌을 때, E_A가 E_0와 isogenous 한지 결정하는 문제.
- CSSI (Computational Supersingular Isogeny Problem) E_0에서 E_A로 가는 isogney를 ϕ_A라 하자. E_A와 $\phi_A(P_B)$, $\phi_A(Q_B)$가 주어졌을 때 R_A를 찾는 문제.

① Classical attack

DSSI와 CSSI 문제를 claw problem으로 치환하면 효율적으로 해결할 수 있으며, 복잡도는 $O(\ell_A^{e_A})=O(\sqrt[4]{p})$이다.

② Quantum attack

Claw problem으로 DSSI와 CSSI 문제를 치환하면 양자 컴퓨터에서는 더 효율적으로 해결할 수 있으며, 복잡도는

$O(\ell_A^{e_A/3}) = O(\sqrt[6]{p})$이다.

<표 2>는 classical attack과 quantum attack을 고려했을 때 SIKE에서 제안한 파라미터와 본 논문에서 제안한 파라미터의 안전성을 비교한 표이다. Cofactor로 인해 본 논문의 소수가 비트 수가 더 크지만, 안전성이 작다는 것을 알 수 있다. 하지만 차이는 4~6비트 정도이며 원하는 안전성을 보장해 줄 수 있으므로 해당 소수 사용이 가능하다.

<표 2> SIKE와 제안하는 파라미터 안전성 비교

종류	Classic	Quantum
SIKE751	2^{186}	2^{124}
OURS	2^{180}	2^{120}

4) 실험 결과

아래의 표는 SIDH 구현하는 데 필요한 단위 연산들에 대한 연산량을 정리한 표이다.

<표 3> SIDH 구현에 사용되는 단위 연산량

	Montgomery	Edwards
Differential Addition	8M+4S	8M+4S
Doubling	4M+2S	4M+2S
Tripling	7M+5S	7M+5S
3-isogeny eval.	4M+2S	4M+2S
3-isogeny coeff.	2M+3S	2M+3S
5-isogeny eval.	8M+2S	8M+2S
5-isogeny coeff.	8M+5S	6M+6S

<표 3>은 제안하는 소수를 사용하여 SIDH를 구현할 경우 Montgomery 곡선과 Edwards 곡선에서의 연산량을 비교한 표이다. <표 3>에서 Alice는 3^{227}-torsion group을 Bob은 5^{168}-torsion group을 사용하는 것으로 한다. 본 논문에서는 isogeny 연산량만 계산하였으며, 타원곡선 연산의 경우 <표 3>에서와 같이 Montgomery 곡선과 Edward 곡선에서 연산량이 같으므로 고려하지 않았다.

<표 4> 제안하는 소수로 SIDH 구현 시 연산량 비교

	Montgomery	Edwards
Alice KeyGen	3,178M +2.043S	3,178M +2.043S
Bob KeyGen	5,376M +1,848S	5,040M +2,016S
Alice Shared Key	1,362M +1,135S	1,362M +1,135S
Bob Shared Key	2,688M +1,176S	2,352M +1,344S
Total	12,604M +6,202S	11,932M +6,538S

일반적으로 F_{p^2}에서 1S=0.8M을 의미하면 Montgomery의 경우 17,565.6M, Edwards 곡선은 17,162.4M의 연산량이 든다. 따라서 Edwards 곡선을 사용할 경우 2.29% 속도 향상이 있으며, Edwards 곡선에 유한체 덧셈 및 뺄셈 연산이 더 적기 때문에 추가 효율을 생각할 수 있다.

4. 결론

본 논문에서는 현재 NIST round 2 후보 중 SIKE의 기반이 되는 SIDH에 대해 w-coordinate를 이용해 구현할 경우 연산량을 비교하였다. 먼저 본 논문에서는 기존 w-coordinate 3-isogney 공식을 최적화하였다. Division polynomial을 사용한 결과 기존 4M+6S에서 2M+3S로 연산량이 감소하였다. 또한, 홀수 차수 isogney를 사용하기 위해 새로운 소수를 정의하였으며 안전성을 분석하였다. 본 논문 결과 Edwards 곡선을 이용할 경우 2.29%정도의 속도 향상이 있었다. 향후 연구로 4-isogeny에 대한 공식을 개발하여 SIKE751 파라미터로 구현할 계획이다.

참고문헌

[1] Bernstein et al. “Twisted Edwards curves,” AFRICACRYPT 2008

[2] Bos et al. “Arithmetic considerations for isogeny-based cryptography,” IEEE Transactions on Computers 68(7), 2019

[3] Castryck et al. “CSIDH: An efficient post-quantum commutative group action,” ASIACRYPT 2018

[4] Costello et al. “A simple and compact algorithm for SIDH with arbitrary degree isogenies,” ASIACRYPT 2018

[5] Costello et al. “Efficient algorithms for supersingular isogeny Diffie-Hellman,” CRYPTO 2016

[6] Farashahi et al. “Differential addition on twisted Edwards curves,” Information Security and Privacy, 2017

[7] Jao et al. “Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies,” PQC, 2011

[8] Kim et al. “Efficient isogney computations on twisted Edwards curves,” Security and Communication Networks, 2018

[9] Kim et al. “New hybrid method for isogney-based cryptosystems using Edwards curves,” IEEE Transactions on Information Theory, 2019

[10] Kim et al. “Optimized method for computing odd-degree isogenies on Edwards curves,” ASIACRYPT 2019

[11] Meyer et al. “On hybrid SIDH schemes using Edwards and Montgomery curves arithmetic,” 2017